Lógica

Resolución SLD

Damiano Zanardini

Graduado/a en Ingeniería Informática
Facultad de Informática
Universidad Politécnica de Madrid
damiano@fi.upm.es

Curso Académico 2010/2011

Introdución

SLD: Selection function in Linear resolution for Definite clauses

- combina la estrategias lineal, input, dirigida y ordenada sobre un tipo particular de cláusulas

Cláusulas de Horn

- como mucho un literal afirmado (y si lo hay, es el primero)
  - $A \vee \neg B_1 \vee \neg B_2$
  - A
  - $\neg B_1 \vee \neg B_2$
- las cláusulas que no tienen literal afirmado forman parte del *conjunto objetivo*
- las cláusulas que tienen literal afirmado forman parte del *conjunto soporte*

Introdución

Definición (resolución SLD)

Una derivación SLD de C_m a partir de un conjunto $\{C_1, .., C_n\}$ de cláusulas de Horn (con el literal afirmado en la primera posición, si existe) es una secuencia $\langle C_1, .., C_i, .., C_n, C_{n+1}, .., C_m \rangle$ tal que

- *C_{n+1} es el resolvente de C_i (cláusula objetivo) y otra $C \in \{C_1, .., C_n\}$*
- *para todo $j > n+1$, C_j es el resolvente de C_{j-1} y otra $C \in \{C_1, .., C_n\}$*
- *cada paso de resolución tiene la forma*

$$\begin{array}{c} L' \vee C' \\ \neg L'' \vee C'' \end{array} \quad \rightsquigarrow \quad (C' \vee C'')(UMG(L', L''))$$

Propiedades: la resolución SLD es

- lineal
- input
- dirigida (para un S concreto)
- ordenada

Resolución SLD

La regla de selección

En SLD la regla determina que el factor de corte sea el primer literal de las dos cláusulas

- como consecuencia, la cláusula objetivo no contiene literales afirmados y se resuelve con otra cláusula cuyo primer literal es afirmado

Por qué cláusulas de Horn

- está claro que no todas las cláusulas son de Horn
- es decir, esta condición *limita* el lenguaje
- sin embargo, dado un conjunto $\mathcal{C}$ de cláusulas, existe un conjunto $\mathcal{C}'$ de cláusulas de Horn tal que $SAT(\mathcal{C})$ si y sólo si $SAT(\mathcal{C}')$

Resolución SLD

Ejemplo: objetivo $C_7 : \neg C(x) \vee \neg E(x)$

$$
\begin{array}{lcl@{\qquad}lcl}
C_1 & : & D(x) \vee \neg A(x) & C_2 & : & E(x) \vee \neg A(x) \\
C_3 & : & A(a) & C_4 & : & B(a) \\
C_5 & : & C(x) \vee \neg D(x) \vee \neg B(x) & C_6 & : & B(x) \vee \neg D(x) \vee \neg C(x)
\end{array}
$$

$$
\begin{array}{c@{\qquad}c}
C_7 & C_5 \\
| & \\
\neg D(x) \vee \neg B(x) \vee \neg E(x) & C_1 \\
| & \\
\neg A(x) \vee \neg B(x) \vee \neg E(x) & C_3 \\
| & \\
\neg B(a) \vee \neg E(a) & C_4 \\
| & \\
\neg E(a) & C_2 \\
| & \\
\neg A(a) & C_3 \\
| & \\
\square &
\end{array}
$$

Resolución SLD

Lema

El conjunto soporte de un conjunto de cláusulas de Horn es satisfacible

Demostración.

1. las cláusulas del conjunto de soporte tienen un literal afirmado
2. una interpretación que asigne **v** a todos estos literales hace verdad al conjunto

Corolario

Si existe una refutación de un conjunto de cláusulas de Horn, entonces existe una refutación dirigida a partir del conjunto soporte

Resolución SLD

Teorema

La resolución SLD es completa para cláusulas de Horn: si un conjunto de cláusulas de Horn es insatisfacible entonces existe una refutación SLD de él

Demostración.

1. $INSAT(H)$
1. existe una refutación de H (por completitud de la resolución)
2. existe una refutación dirigida $\mathcal{R}$ (el conjunto soporte es satisfacible)
   - en cada paso interviene una cláusula objetivo o un resolvente intermedio
3. $\mathcal{R}$ es una refutación input
   - cada paso requiere una cláusula con literal afirmado: una cláusula soporte
   - las cláusulas de soporte son cláusulas de entrada
4. si existe una refutación input existe una refutación input lineal $\mathcal{R}'$
   - $\mathcal{R}'$ es dirigida, input y lineal
5. existe una refutación SLD $\mathcal{R}''$ (lema no visto)

Resolución SLD

Al estudiar un conjunto de cláusulas de Horn

- las refutaciones posibles se pueden restringir a refutaciones SLD
- los árboles de búsqueda se puede restringir a árboles SLD para $\Box$

Anchura y profundidad

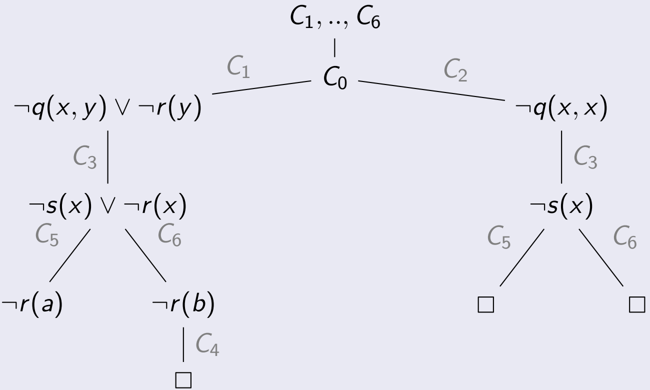
- SLD en anchura es completa, en profundidad no lo es
- en la búsqueda en profundidad es muy importante el orden en que se eligen las cláusulas soporte que se pueden resolver con la cláusula objetivo en curso
  - *función de computación*
- dependiendo de la estrategia de búsqueda
  - algunas refutaciones no se hallan
  - algunas derivaciones no finalizan

Resolución SLD

Ejemplo

$C_1 : p(y) \vee \neg q(x,y) \vee \neg r(y) \qquad C_2 : p(x) \vee \neg q(x,x) \qquad C_3 : q(x,x) \vee \neg s(x)$

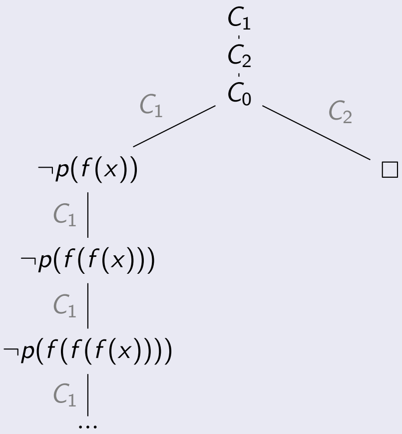
$C_4 : r(b) \qquad C_5 : s(a) \qquad C_6 : s(b) \qquad C_0 : \neg p(x)$



Resolución SLD

Ejemplo: $C_1 : p(x) \vee \neg p(f(x))$, $C_2 : p(a)$, $C_0 : \neg p(y)$

- una búsqueda en profundidad con una función de computación que elija la primera cláusula soporte no termina



Resolución SLD

Ejemplo: $C_1 : p(a)$, $C_2 : p(x) \vee \neg p(f(x))$, $C_0 : \neg p(y)$

- pero una refutación se puede obtener cambiando el orden de las cláusulas soporte (cambiando C_1 con C_2)